

DOCUMENT RESUME

ED 253 213

IR 011 489

AUTHOR Dalgish, Gerard M.
 TITLE Microcomputers and Teaching English as a Second Language. Issues and Some CUNY Applications. Research Monograph Series Report No. 7.
 INSTITUTION City Univ. of New York, N.Y. Office of Academic Affairs.
 PUB DATE Sep 84
 NOTE 58p.
 PUB TYPE Information Analyses (070) -- Viewpoints (120) -- Reports - Evaluative/Feasibility (142)
 EDRS PRICE MF01/PC03 Plus Postage.
 DESCRIPTORS *Computer Assisted Instruction; *Courseware; *English (Second Language); *Evaluation Criteria; Evaluation Methods; Higher Education; Instructional Materials; *Microcomputers; Program Descriptions; Second Language Instruction; Word Processing
 IDENTIFIERS City University of New York; Software Evaluation; *Software Reviews

ABSTRACT

This review focuses on microcomputer software designed to improve the writing skills of students of English as a Second Language (ESL) at the college level. It presents a case for the use of computers in teaching ESL, provides guidelines and suggestions for evaluating or creating software and programs, and reviews representative software, including materials from the Minnesota Educational Computer Consortium, Regents/ALA, BIPACS, Dormac, Computer Curriculum Corporation, Hartley, Teacher's Friend, and PLATO. The applicability of such word processing packages as Bank Street and Franklin AceWriter to the ESL class is addressed. Quality control in ESL software development is urged, and methods of achieving such control are outlined. Several specific computer assisted instruction (CAI) projects currently in the planning stage or already underway at the City University of New York are specifically described, as well as research on the use of computers in teaching ESL. An informal glossary of computer terminology and a list of sources and resources are included. (LMM)

 * Reproductions supplied by EDRS are the best that can be made *
 * from the original document. *

ED253213



U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)

✧ This document has been reproduced as
received from the person or organization
originating it.
Minor changes have been made to improve
reproduction quality.

- Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

RESEARCH MONOGRAPH SERIES
REPORT NO. 7

**MICROCOMPUTERS AND
TEACHING ENGLISH AS A SECOND LANGUAGE**

ISSUES AND SOME CUNY APPLICATIONS

September 1984

Instructional Resource Center
Office of Academic Affairs
The City University of New York
535 East 80th Street
New York, New York 10021

"PERMISSION TO REPRODUCE THIS
MATERIAL HAS BEEN GRANTED BY

Marie Jean Lederman

IR 011489

THE CITY UNIVERSITY OF NEW YORK

BOARD OF TRUSTEES

James P. Murphy, *Chairperson*
Edith B. Everett, *Vice Chairperson*

Paul P. Baard
Blanche Bernstein
Sylvia Bloom
James Cavanagh
Louis Cenci
Armand D'Angelo
Judah Gribetz
Harold M. Jacobs
Jules Kolodny
Melvin E. Lowe (*Chairperson, University Student Senate*)
Robert L. Polk
Joaquin Rivera
Margaret Titone
Henry Wasser (*Chairperson, University Faculty Senate*)
Brenda F. White

Joseph S. Murphy, *Chancellor*

THE COLLEGES

Baruch College—Joel Segall, *President*
Borough of Manhattan Community College—Joshua L. Smith, *President*
Bronx Community College—Roscoe C. Brown, Jr., *President*
Brooklyn College—Robert L. Hess, *President*
City College—Bernard W. Harleston, *President*
City University School of Law at Queens College—Charles R. Halpern, *Dean*
Graduate School and University Center—Harold M. Proshansky, *President*
Hostos Community College—Flora Mancuso Edwards, *President*
Hunter College—Donna E. Shalala, *President*
John Jay College of Criminal Justice—Gerald W. Lynch, *President*
Kingsborough Community College—Leon M. Goldstein, *President*
LaGuardia Community College—Joseph Shenker, *President*
Lehman College—Leonard Lief, *President*
Medgar Evers College—Jay Carrington Chunn, II, *President*
Mount Sinai School of Medicine (affiliated)—James F. Glenn, *President*
New York City Technical College—Ursula C. Schwerin, *President*
Queens College—Saul B. Cohen, *President*
Queensborough Community College—Kurt R. Schmeller, *President*
The College of Staten Island—Edmond E. Volpe, *President*
York College—Milton G. Bassin, *President*

**MICROCOMPUTERS AND
TEACHING ENGLISH AS A SECOND LANGUAGE**

Issues and Some CUNY Applications

Gerard M. Dalgish
ESL Supervisor
Department of English
Baruch College
New York, N.Y.
212/725-7105

Acknowledgements

I wish to thank my colleagues at The City University of New York for their helpful comments and suggestions in the preparation of this study: John Brandeau, Irene Dutra, Peggy Eberbach, Jason Kim, Ellen Leahy, Herbert Seliger, Michael Southwell, Dave Tilyer, and Dianne Ujhazi. Special thanks also to Professor Mary Hiatt, Dean Martin Stevens, and Assistant Provost Robert McDermott of Baruch College for their support of my research.

For reasons I have never understood, it is customary to mention last those who have helped the most. My editor, Virginia Slaughter, and her assistants at various stages, Robbin Juris and Karim Feijoo, worked with me in the early stages and were tireless in the preparation of the final manuscript. Any errors or omissions that remain, in spite of their exacting scrutiny, are my responsibility. Finally, I would like to thank Dean Marie Jean Lederman who is largely responsible for my growing involvement in computers at CUNY. Her guidance, support, and encouragement not only made this study possible but helped me clarify my vision of microcomputers in education.

Gerard M. Dalgish

July 10, 1984

TABLE OF CONTENTS

	<u>Page</u>
I. PURPOSE	1
II. OVERVIEW	4
III. ESL SOFTWARE: WHAT IS AND WHAT COULD BE	9
IV. SOME CUNY CAI PROJECTS	29
V. CONCLUSION	36
VI. GLOSSARY	39
VII. REFERENCES	46
VIII. SOURCES AND RESOURCES	49

I. PURPOSE

Anyone interested in computer-assisted instruction (CAI) in English as a Second Language (ESL) at the college or university level will be somewhat dismayed at the lack of suitable materials presently available. This is ironic in light of Dodge's (1980) comment that perhaps a majority of institutional CAI language-instruction material is in ESL. Many institutions, as well as commercial book publishers, have begun to flood the market with ESL software, but most of it is aimed at the K-12 level. This review will not concern itself primarily with these materials, but will focus instead on computer software designed to improve the writing skills of ESL students at the college level.

In this study I wish to make a case for the use of computers in teaching ESL, provide guidelines and suggestions for evaluating or creating one's own software and programs, and review some of the most representative software, including materials from MECC (Minnesota Educational Computer Consortium), Regents/ALA, BIPACS, Dormac, Computer Curriculum Corporation, Hartley, Teacher's Friend and PLATO. I will briefly discuss the applicability of certain word processing packages such as Bank Street and Franklin AceWriter to the ESL class. I will urge quality control in ESL software development and outline ways of achieving it. In addition, I will discuss some of the CAI projects at The City University of New York (CUNY)--some in the planning stage and others already under way. I will also discuss

some ESL research utilizing computers, an area in which the possibilities are enormous.

An informal glossary of computer terminology appears in the back of the report along with a list of sources and resources.

This paper is modeled on Joan Baum's study (1983), No. 6 in the Research Monograph Series published by the Instructional Resource Center (IRC) of CUNY, which focuses on computers in the English class. Her study looked at pre-writing, poetry, and some other areas that are not the primary concern of ESL pedagogy. This is not to say, of course, that such areas are of no interest to ESL faculty, merely that our major emphasis is elsewhere. Some word-processing software will be mentioned, but anyone wishing to read more on that topic should see Baum's work, as well as McWilliams (1982) and a forthcoming review and analysis of selected word processing packages by Brian Gallagher (in press, 1984).

Another similarity to Baum's study is the method I used in surveying those people at CUNY interested in computers and ESL. Phone calls to directors of ESL departments were followed by visits to labs whenever possible; notices were sent to interested parties, and calls for information were published in the CUNY ESL Council Newsletter and announced at meetings of the CUNY ESL Council. Leads were followed from one contact person to another, but eventually all contacts led back to a small circle of people who were actively engaged in CAI. Most were at the planning stage in developing CAI labs or software, and many shared the views of Nancy Duke Lay (City College) who feels we

are just beginning to break ground in the use of CAI and ESL and that the field is still so young that we have not yet formulated a clear sense of direction. Many CUNY ESL faculty acknowledged that they do not feel comfortable with computers. They wanted to be trained in the use of computers before venturing into training students to be "English-literate" via computers. This precaution is eminently wise given the CAI bandwagon that is careening along with little or no direction.

II. OVERVIEW

The acronym GIGO (gi/go) is computer jargon for the idea that if the computer is given poor, useless input, that is exactly what it will return (this phrase is more than relevant when it comes to ESL CAI). The acronym dates back at least to 1966, but its referent, the phrase "garbage in, garbage out," probably is somewhat older. One hears quite often that the available courseware in ESL is primitive, dated, restrictive, lock-stepped, and frequently does not exploit the full potential of the computer. More on this later. But given these deficiencies, why would someone advocate CAI in ESL?

My answer assumes certain basic premises about what kind of computer assisted instruction can best serve all students and faculty, including those in ESL. Some of these premises involve correction without (negative) criticism, immediate feedback (sometimes in the form of graphics/animation), and student autonomy, control, and interaction. These ideas will be discussed later in conjunction with an examination of criteria for ESL courseware. For now, I wish to mention some additional considerations favoring the use of CAI in ESL.

The possibilities for individualized instruction in CAI have enormous potential for application in ESL. Restricting our discussion to written grammatical errors, the range of possible error types for basic writing and ESL students is considerable. Within such error types, there are sub-types (verb tense would be

a general type and the use of modals within that type would be a subtype). Given the many possible error types, there is a great need for individualized instruction for ESL students, including computer assisted instruction; but unfortunately, that is missing in most available courseware.

If you add to this the fact that certain grammatical error types are more likely among certain linguistic groups of ESL students (due to varying degrees of first language interference), the need for individualization along linguistic lines becomes even more critical. Yet this aspect of individualization is also lacking in current software.

Computers could even circumvent a common objection to the idea of having ESL classrooms of students from one linguistic background, namely, that such students would be over-dependent on their first language, either by deliberately or subconsciously conversing in it during discussions. With CAI individualized by language, the class itself can be linguistically heterogeneous, while the instructional material can be targeted to the student's first language. With the computer, the teacher can avoid being the enforcer of the rule "English only please," and the "heavy" who negatively criticizes mistakes.

Similar arguments could be made for the use of CAI in developing American English rhetorical patterns for ESL. Certain rhetorical methods of organization of the student's first language often interfere with or are in direct contrast to the generally accepted English pattern of introduction and thesis, topic sentence in the main body paragraphs, development from the

general to the specific, etc. (Kaplan, 1970). ESL courseware individualized to a student's first language rhetorical patterns could be of great assistance. Admittedly, the need here in the United States, and specifically in the CUNY system, for such courseware may not be economically justifiable, largely because students have less difficulty absorbing the American English rhetorical style than they do in absorbing American English grammar, vocabulary and idiom. Still, the potential exists for instruction aimed at individual ESL students' problems. Another potential is software that emulates word-processing to help students see more readily how their organizational patterns differ from the English model and how readily their essays can be adjusted to conform to that pattern. This would encourage students to believe that it is often not their ideas but simply their patterns of presentation that are being criticized. Using the computer to manipulate readily the elements of the essay can help students improve their presentation of ideas and build up their confidence in writing.

The potential for graphics and animation in ESL grammatical instruction is important as well. Certain troublesome grammatical concepts are extremely hard to grasp for ESL students and may only be partially understood by ESL teachers themselves. An appropriate example is the full range of deceptively simple English articles. For example, "I see a dog" vs. I see the dog;" "Close a window" vs. Close the window;" "The United States" vs. "America;" (not *"The America"). Linguistic/grammatical explanations may also be only partially comprehensible by both

parties, and exercises from textbooks with isolated examples can only be of minimal usefulness in many cases. It is often disconcertingly clear that our explanations have been misunderstood, wrongly interpreted, or overgeneralized. A non-language, visual image of the grammatical concept provided by the computer can create the correct impression and can be more easily recalled and applied than technical terminology. This imaging concept for reinforcement should neither be mere cartoons or animated comic strips that tend to insult the intelligence of the user, nor the slow, skeletal manipulation of morphemes that is reminiscent of the old video game, PONG, that also characterizes the sparse animation/graphics common today. Specific suggestions about the effective use of animation/graphics will be discussed in the next chapter, but for now it should suffice to say that the explanation of complex, semantically-based grammatical concepts could indeed benefit from CAI, and it is precisely these complex concepts that need to be explained in the ESL classroom. Of course, the more mechanical rules of English writing (agreement, parallelism, punctuation) may also benefit from graphics explanations, but it is the experience of most college-level ESL teachers that these mechanical troublespots are not what stigmatize students whose writing errors are labelled "ESL."

The computer in an ESL class can do more. In addition to encouraging literacy and precision (if commands are not exactly right the computer will not work), it may instill the feeling that not only is the computer "logical," but so is much of

English grammar. Good ESL courseware can help students see the patterns of their errors, understand why the errors were made, and learn a skill--the manipulation of a computer--that has real-world application.

There has, in fact, been an awareness of the potential of CAI in ESL for some time. The TESOL Quarterly (Teaching English to Speakers of Other Languages) and its newsletter "welcomes evaluative reviews of...computer and video software," and there are numerous symposia, workshops, and presentations given at the annual conventions. A Special Interest Group on CAI within TESOL was formed in 1984; a similar group was formed earlier within the Conference on College Composition and Communication (CCCC). The New York State affiliate, NYS TESOL, has held conferences with CAI presentations. CUNY and Long Island University co-sponsored a conference on "Microcomputers and Basic Skills" (a publication based, in part, on the proceedings of that conference is available through the IRC), and an ESL component featured prominently in that conference. CUNY sponsored a national conference on the same theme in April 1984. ESL CAI is also alive and well internationally. Among others, the British have produced some very useful software called WIDA, Ltd.

In addition, the subject matter and its target (the ESL student), are well suited to the capabilities of CAI. The question is not whether, but when, to adopt some form of CAI in ESL. The major stumbling block has been the availability of appropriate software, and it is this topic that I will address in the following sections.

III. ESL SOFTWARE: WHAT IS AND WHAT COULD BE

In this chapter I will discuss the characteristics of some college level ESL software available at the time of this writing. Instead of an in-depth review of each individual piece of software, I will address the deficiencies that recur in much of it. In the second division of this chapter, I will offer some suggestions for improving the quality of ESL software.

As a reviewer, I was torn between the need to point out the deficiencies and at the same time to soften the criticism with explanations for those deficiencies. After all, it is relatively easy to discount much of the software because it is of very low quality. There are, however, some mitigating circumstances -- high production costs, piracy problems, consultants' fees, and technical programming difficulties. But these deficiencies should not be too hastily forgiven; the costs and the stakes are too high for us to be too soft on software. The circumstances--which all boil down to cost--simply must be taken into account when evaluating and improving software. I will discuss some of these circumstances in the second section of this chapter in my discussion of what ESL software could be.

It is clear, however, that the cost of developing high quality software and the uncertainty of the future directions of the CAI market for ESL contribute to the publishers' reluctance to take economically unfeasible risks in investing the necessary

time and expense to produce better courseware. First let us turn to a discussion of the current state of the art of CAI.

What Is

When I began this study, I contacted City University faculty interested in ESL and CAI for their reactions to ESL software. My research in large measure confirms their views. Much of what they said can also be a general criticism of CAI. Herbert Seliger (Queens College) commented that the poor quality software flooding the market was "re-inventing the 60's" with language-lab type materials based on Skinnerian stimulus-response (S-R) strategies that bored the students. To a large extent, these observations are a result of the fact that CAI authoring languages in which most courseware is written were produced by practitioners of programmed instruction, a method of pedagogy based primarily on the S-R behaviorist learning theory. In other words, it is the nature of CAI programming to be behaviorist, right down to the computer language used to produce the courseware. Experts and novices alike might say that computers will, because of their nature, always be restricted to S-R-type interaction. ESL teachers shun S-R type materials and methods so it should be no surprise that most ESL software created in this vein has met with disfavor.

Another general CAI shortcoming of ESL courseware is that it is restricted to drill and practice in which the screen is the rough equivalent of the textbook. This is bad enough when courseware consists of some text/instruction and exercises (as in most ESL software), but it is even worse when the "textbook"

resembles, in form and content, no more than ditto sheets as in the MECC Teacher Utility mini-authoring software. Most observers agree that there is no point to this use of CAI (Bork 1981, The 1982, Baum 1983, inter alia). Of course, the relative value of the textbook would affect the quality of the software (GIGO again), but even an excellent textbook reproduced on the screen would be expensive and of little value.

In ESL software there is very little departure from an exercise book. Even when the instructional text of the CAI courseware is of fairly high standard (notably, the Regents/ALA ESL material), the problem of not utilizing the full potential of the computer surfaces again. The result is that lesson material is bookish and boring, and students, faculty, and reviewers quickly lose interest.

This is the major criticism of ESL software, and holds for most of what I reviewed. For example, BIPACS, a software publishing company, has some slow, low-level graphics involving words which appear one by one in cartoon bubbles over static characters, and some grammar lessons include movement of certain words; but this is slow and not too gripping. Another--PLATO's remedial grammar lessons--shows morphemes or words scooting around the screen as a result of various transformational rules. Although PLATO is probably the best of the institutionally available courseware in terms of its utilization of the graphics capabilities of the screen (graphics, speech, and touch-screen capabilities), these lessons are already nearly a decade old, and the computer is currently capable of much more. For example,

Irene Dutra (Bronx Community College), using limited memory capacity, has developed lessons on past tense spelling rules which are already a step beyond PLATO's systems in terms of creative use of graphics. Some of Michael Southwell's (York College) Comp-Lab exercises, though not necessarily ESL, show creative use of color and animation for explanations and corrections of student exercises. On the other hand, I found that Regents/ALA, a software publishing company, has a chapter, "Expression of Quantity" which was disappointing because the presentation of the material was too static although it was clear, with a nicely constructed grid showing which nouns can go with which articles/determiners, etc. The drawback is that the grid could just as easily have been in a textbook. Why not have nouns or articles/ determiners float into their proper slots, or even better, have the students manipulate them to match each other correctly? This is a case of grammatically sound lesson content failing to be matched--imaginatively or not--to the capabilities of the computer. The Regents/ALA material seemed the best in terms of grammatically useful, correct, and sophisticated instructional courseware, including a good set of dialog questions and answers; however, it could have gone further in terms of computer potential.

Another major criticism of the software available in ESL is the lock-step quality of the lessons. There are at least two subdivisions of this shortcoming: inability to move forward or backward within a lesson, and inability to skip ahead to another lesson. The PLATO ESL software (University of Illinois and

University of Delaware versions) is notorious for this shortcoming. In PLATO one must go through and answer every question and exercise before one can be "released" to go on to another. In other ESL packages (Regents/ALA, DORMAC and BYPACS), one cannot go back to an earlier part of the lesson without removing the diskette, booting up from scratch, and answering all the questions again. Regents/ALA, unlike most software, provides a review of incorrect answers; the wrong answer or one like it is put before the student again at the end of the lesson. This is highly desirable, although one still cannot go back to find the principle or rule behind the wrong answer, that is, without booting up again. As in PLATO, the student cannot go on until he gets the right answers. Furthermore, the courseware is not programmed to detect or to demonstrate to the student that these errors form any sort of pattern. One error after another is flashed on the screen, taken in numerical order from student mistakes. This lockstepping problem is related to the program's lack of branching capability. Since there is no distinction between a typo and a wrong answer, the student is forced to slog through correctional exercises which he may not need and without learning the basis for his errors.

Student control of the computer lesson is an important quality of good CAI, but sometimes the ESL lessons "take over." The Regents/ALA Grammar Mastery introductory diskette added paragraphs and "turned pages" without specific instructions from the student (they were automatically timed); in most of the later sections, however, notably the information lessons on Expressions

of Quantity, some control was returned to the student. At the University of Illinois the PLATO system in some lessons enforces time rules and bumps the student off the computer.

Students should feel that they control the computer, not the other way around. If a student inputs an error of a trivial nature--that is, a mere typo on a word not related to the target structure--and/or the computer gives an inappropriate response (cf. below), these will almost certainly promote a feeling of powerlessness over the computer which can easily be transferred to the instructional material itself. In addition, students often miss a chance to manipulate or control the computer in ways that go beyond the typing in of correct responses. While I am not advocating arcade game lessons, there should be room for student control over graphics-type lessons.

The amount of student input, or lack of it, is part of this concept of control and another important criterion for evaluating the usefulness of ESL software. For this category there is once again a wide range of unacceptability. In some software, lessons are nearly entirely multiple-choice, in which the student's input is not even a word, phrase, or sentence, but a correct letter or number. Teacher's Friend uses multiple-choice questions in which the student types in the number; it also has true/false grammar questions. Hartley has within the same lesson multiple-choice questions in which the letter of the correct answer must be chosen; later in the lesson the word itself, taken from a choice of words, must be typed.

Some DORMAC syntax drills are multiple-choice by letter--a, b, c--no student interaction here. A failing in these sets is that even after the correct answer is chosen, the student does not see his completed, correct sentence with the correct choice put in its proper place. He is merely told "That's correct," and is given the next sentence, where the same thing happens. Minimal student input like this does little for recognition of correct answers, but does even less for learning how to write well. As Dutra says, many of these lessons neither "bring or assume" any intelligence to or of the student.

Some PLATO (Illinois) lessons go to the other extreme in requiring lengthy input that is more subject to typographical errors (see Stevens 1983). The result is that the student's sentence is classified as incorrect even though his grammar is correct. A typo on a word unrelated to the syntactic target of the lesson renders the sentence incorrect. Because of the inappropriateness of the computer response, the student does not know what he has done.

In terms of student input, then, most ESL software demands too little or too much. For the ESL student, too little input means the loss of an opportunity to practice production; too much required input dampens enthusiasm for creativity or promotes an "only-one-answer-is-correct" mentality that is too confining. ESL teachers spend enough time trying to encourage students to vary their writing styles. Why have the computer do the opposite?

Another sub-category of the problem of student control concerns the appropriateness of the computer's response to student input. Inappropriate responses run the full gamut from annoyingly "user-friendly" messages like "Way to Go," and tonal beeps from "victorious" to "agonizing," through such bland and essentially useless answers like "That is not the correct answer," "Try it again," (Teacher's Friend) that give no reason for the error, to downright wrong responses that tell a student his answer is right or wrong when in fact it is not. Since the first category is really a question of taste, I will not concentrate on it beyond saying that sound effects can sometimes be controlled or removed, especially in a learning center or library setting. These video arcade effects do seem out of place at a university and only serve to detract from the seriousness of the educational computer.

Nothing is more useless than being told one has given the incorrect answer without being told why. The reasons for lock-stepped progress through lessons, lock-stepped input, and lock-stepped computer responses have to do with the lack of branching capabilities. Branching is a term used to describe a type of programming that is capable of responding in different ways, depending on varying student input (it sometimes refers to skipping ahead in a lesson, but this, too, depends on student input). Much software, in particular ESL software, has very limited branching. Typos, misspellings, nonsense words, and gibberish may all be lumped together by the computer as wrong answers, alongside genuine errors, but together with

abbreviations, contractions, or rephrasings that are essentially appropriate or correct responses. This is because a blanket computer response is often programmed in for every input that does not match a single answer. Peggy Eberbach (York College) relates an instance when reviewing the PLATO (Delaware) lessons on articles, where she typed "shes" (no apostrophe) and was reminded that "the article goes before the noun." Typing in "aren't" instead of "are" in PLATO, Exercise One (Be Going), results in being told that that is the wrong form of the verb "to be." But even when the student makes a genuine error on some multiple choice type exercise, the computer response does not explain the reason for the error. Software guilty of this includes DORMAC, Teacher's Friend, and Hartley, although DORMAC will sometimes give the entire--not individualized--explanation. The student may be given a few chances to get it right, and may sometimes be told the correct answer, but there is little or no connection to the cause or the rule behind the correct answer. By allowing one--and only one--correct answer, the lesson stifles creativity in language use. Variation and creativity, which adequate branching would facilitate, are the very skills that ESL students need encouragement to develop and use, since so often their writing is restricted both in syntax and vocabulary.

Some ESL software lessons keep and provide a "score" of the student's performance on the lesson, which might be useful to the teacher interested in such tallying. The Regents/ALA Grammar Mastery software allows a teacher to read a student's scores from lesson diskettes, provided the teacher gets the diskette before

the next student uses it. The degree of availability and ability of score-keeping varies, but an important problem persists: most programs do not allow the computer's response and the student's response to be printed out. Although PILOT is probably the most popular authoring language, at the present time it has no provision for reproducing the contents of the session as hard copy. The result is that the teacher cannot determine where the student went wrong, why he went wrong, if there is a bug in the computer, or if the student attempted the right answer but entered it the wrong way--perhaps by hitting the space bar instead of the return key. To my knowledge, most of the ESL commercial software has limited capability for hard copy of student input and computer response beyond multiple-choice.

Not only does the student not have much control, but in most ESL software there is little opportunity for the teacher to add to or modify the existing programs. In fact, teachers cannot copy or modify the software they purchase because most of it is "locked" or protected electronically to prevent pirating. Some lesson software may claim to be adaptable, but the extent to which the lessons can really be altered is usually insufficient, at the most changing vocabulary input. The MECC English mini-authoring program allows a teacher to select sentences from within the disk or lesson and to write some words or sentences, but this is limited. Unless the ESL teacher is prepared to write his own software (authoring), control of the lesson is not in his hands.

Another problem that arises with locked software concerns factual errors in the text or lessons. The Hartley series gives us such news as "The past tense of an irregular verb is often the same for singular and plural verbs." (!) "The verb in English has three main forms; [sic] infinitive, past, past participle." This kind of misinformation cannot be corrected or removed by the teacher because the software is protected. Since most of what appears on the screen is believed to be written in stone--and if locked may as well be so written--the consequences of factual error in software are grave.

Documentation (computerese for the reference guide accompanying the software) suffers from inadequacies within and outside ESL. Often the documentation is disorganized, badly sequenced, poorly indexed, unnecessarily colloquial, but most importantly, lacks explanations of what one needs to know--for instance, why the system is down or won't boot up. Moreover, HELP menus are hard to access, and are often incomprehensible even when found. Most teachers, unless they have extensive computer training, will feel intimidated when they cannot fix what is wrong by reading the documentation. Most ESL students, in turn, will be discouraged.

In general, if one is interested principally in drill and practice (S-R), there is much material available, and it is relatively easy to create one's own "lessons" in this framework. This use of the computer has certain value as an adjunct to classroom instruction. It is, however, limited both in the quality of existing material and in the fulfillment of the full

potential of computer assisted instruction for ESL students. There is need to explore and create new uses for microcomputers in the ESL classroom.

What Could Be

As present, there are at least four possible functions of CAI: (1) task master, (2) tool (as in word-processing), (3) tutor (instructional text), and (4) tutee (in which the student programs the computer). Much of the existing ESL software could be likened to a modulated, repackaged; "New Army" drill sergeant, structured in the S-R mode. And there is some useful software that is applicable in ESL as a tool for writing, notably the Bank Street and other word processing software. Though not specifically designed for ESL students, the Bank Street word processor has been used successfully in ESL classrooms. I believe that similar success could be had with Franklin's AceWriter II, a simple yet powerful word processing package similar to AppleWriter. And some ESL colleagues report success with Homeword, word processing software that uses icons for many commands (an arrow pointing to a wastebasket signifies deletion, for example).

ESL CAI needs improvement in the field of computer as tutor. The computer language LOGO--a high level, easily acquired language--is already being used in some classes to turn the computer into a tutee. Dutra has her more advanced students prepare and program CAI lessons for other students, thereby

encouraging precision and helping them learn the real-world skill of computer programming.

Ellen Leahy from Bronx Community College, (Learning Specialist), agrees. A good CAI program, she believes, meets a specific world need; in this case, it is the appropriate use of language. She favors and has helped implement CAI curricula in which there are "short, regular trips to the computer."

Most faculty would agree with Leahy that good courseware should not stand alone; it should reinforce classroom instruction. Good CAI should match the plan of the curriculum: If instruction is in the class, it should not be repeated at length on the computer. In like manner, concepts should not be explained in a totally different way from one medium to the other. The content of CAI, even if it is just drill and practice, must relate to classroom work, even if the two are alternative methods of presenting that work. It is not, however, a good idea to model CAI lessons too closely to any one text (Marty, 1981).

Good CAI material should replicate good teaching. It should be reactive (teacher responds to and adapts to student needs) and interactive (students don't just watch or listen; they do). Ideally, there should be much more branching, allowing students to take different paths through the lessons as their input and understanding differ. Effective software should also allow for varied computer responses and answers with explanations to varying student input. Further, CAI that allows for full student participation through the lessons fosters a sense of control over

both the computer and the language students are learning. They should be able to go forward and backward within a lesson, and teachers should have easy access in order to adapt their lessons to their students' needs.

ESL CAI should also fully utilize the potential of the computer, employing visually stimulating or striking graphics and animation. The recent development of integrated video--a videotape player or videodisc player controlled via an interface with the computer--has great potential in ESL CAI. Videotaped material, or material recorded on videodisc, can be integrated with regular printed lessons on the computer. The advantage of this mode of presentation is that a number of grammatical/semantic concepts that trouble ESL students could be graphically illustrated on the computer (tense, mass/count distinctions, and the syntax and semantics of difficult verbs like concern or involve, say vs. tell, and others). Grammatical rules and explanations without such graphic material have their place, but there are shortcomings, most notably that the conceptual explanations are most often couched in the language that is being taught. Try to explain, for example, the concept of the present continuous tense without using it in the explanation. The capabilities of the graphics and interactive video of the computer can transcend the metalanguage of the grammar and go a long way towards projecting the image of the grammatical/semantic concept itself without obfuscating the concept with jargon. And this would obviate the danger of presenting the material in the student's first language as is done in the Computer Curriculum

Corporation ESL courseware, where grammar and vocabulary are presented in Chinese and Spanish. Instead, this "notional-imagery" concept is a step beyond the more traditional CAI ESL method where words or morphemes move around the screen as a result of various linguistic grammatical transformations. I think the latter is useful, but again, limiting. One does not imagine that students form a WH- question--a question starting with Who, What, When, Why, Where--by questioning a noun phrase, inserting the proper WH- form, and fronting it with auxiliaries trailing behind as in PLATO (Illinois). This type of grammatical/syntactic imagery may have its place, but the more vivid integrated mode of presentation is potentially more effective for reinforcing the syntactic/semantic concepts which are troublesome for ESL students.

In addition, CAI materials should allow the teacher to obtain hard copy of the student session, in order to discover not only where the most problematic grammar areas are, but also to determine if the student is manipulating the computer correctly (he may understand the grammar, but not the computer). The teacher has time to look for a pattern of errors and make the individualization of instruction a reality by focusing only on the areas in which the student needs help.

The software should be individualized not only to the student's writing problems (rhetorical and grammatical) but also to problems stemming from his first-language interference. Since languages employ different methods of essay development (Kaplan 1970), ESL CAI on rhetoric could be individualized according to

the student's first-language. Lessons could be devised contrasting the differences between Chinese and English essay development; after an introduction to the general concept, the student presses "C" if his first language is Chinese to branch into his language-specific lesson. The same concept is applicable, and probably more useful, to grammar instruction. My research (Dalgish 1983a) indicates that certain types of errors occur within language groups, and that patterns of errors emerge in large-scale studies. The omission of "a," the indefinite article in the writing of Chinese-speaking students is one example; another example is the confusion of simple present and present continuous verb tenses in the writing of Russian-speaking students. When these patterns are isolated, language-specific individualized grammatical instruction can be devised.

Documentation should be clear, useful, and written in correct and concise English. User-friendly (see Glossary) documentation, if overdone, (Franklin Computer Corporation's documentation is a case in point) gets tiring and takes too long to explain to the student how to use the software or why it will not work. Often documentation is misspelled and sentence structure is needlessly complex, full of fragments, and loaded with pronouns whose referents are often indeterminate. HELP Menus (instructions on the screen or in the documentation that direct the user to a set of commands) should contain mnemonic abbreviations that can be learned easily without frantic thumbing through an indexless manual. Moreover, a thorough index together with a glossary of plain English and computer terms should be a

part of ESL courseware documentation. It should obviate the need for extensive computer training of students and faculty.

"Learning overhead time" should be kept to a minimum; good documentation and clear HELP mechanisms can go a long way toward accomplishing this goal.

Educators would embrace the development of a new authoring language or system combining the powerful features of the well-known computer languages used in authoring--BASIC, LOGO, LISP, and PASCAL--with the ease and simplicity of packages like PILOT which are usually too restrictive and not versatile enough. The development of such an authoring language or system would to a great degree reduce the exorbitant amount of time and cost of producing CAI lesson materials. Time is the most important factor responsible for the lack of quality software. Estimates vary widely on the number of hours of computer-programming time it takes to produce one hour of CAI lesson material, but all are exorbitantly costly in terms of time: Otto (1980), 150 hours; Burke (1982), 200 hours; Irene Dutra, between 500 to 1000 hours; Herbert Seliger, quoting an IBM statistic, between 5000-6000 hours. These estimates vary both because of the differences in the software subject matter and because of the differences in the amount of branching, use of graphics, etc. In any event, if one such authoring language could be developed, CAI courseware development would accelerate as costs come down, the computer manufacturers would have a larger market, and CAI educators would have more of a chance to provide meaningful input to the courseware. And if this language could be made compatible to the

computers on the market, we might avoid finding ourselves at the foot of the Tower of Babel, with computers incompatible with each other, and sometimes incompatible with models of the same manufacturer.

To realize some of these improvements, it is crucial that more educators become involved in the production of CAI software. We need a consortium of educators, programmers, and publishers to produce better software, and not just for the ESL field. Educators are sometimes consultants in name only; they may have their names on software or in catalogs, but actually have had little significant input into the design and production of the materials. What we need instead is for the ESL educator and the programmer (each with varying degrees of skill in the other's field) to work together rather than for the educator to produce the text of the lesson, submit it to the programmer, and not see the product until it is published. I know from personal experience that this occurred at times during PLATO lesson development at the University of Illinois. Judging from a favorable review of ESL PLATO CAI in TESOL (Stevens, 1983), perhaps this is no longer true. But the ESL CAI on the market suffers from limited input from educators, despite advertising claims to the contrary.

Failing the formation of a consortium, publishers could hire and train educators or their own personnel to become consultants to other ESL educators, working at each step of the CAI process: to design the curriculum, to develop the physical facilities (advise on hardware choices, for example), to help with software,

documentation, or research writing and development, and to assist and advise educators in the day to day operation of the CAI facility.

In summary, I have examined many of the shortcomings found in the current, limited crop of ESL software and have proposed some guidelines for improving it. The conclusion is that much of the material is inadequate, especially at the postsecondary level, and that unless some steps are taken toward implementing a consortium of educators and software publishers working together, we will be treated to more of the same. At CUNY, there are already some promising signs in this direction.

IV. SOME CUNY CAI PROJECTS

Because it is important within our university to exchange information on the use of the microcomputer for ESL instruction, this chapter will focus on descriptions of ongoing and proposed programs within CUNY. This need for networking was the overriding concern of participants at the ESL Special Interest Group meeting at the 1984 "Microcomputers and Basic Skills in College" conference sponsored by CUNY's Instructional Resource Center, as more and more schools were about to initiate CAI programs. Participants wanted practical suggestions about what hardware and software to use, where to turn for help, and what problems had been encountered by others who already had programs. Many conferees were anxious to keep in touch with each other, and so, as ESL Committee Chair, I mailed a newsletter, address list, and bibliography to all participants.

I hope that the following picture of the stages of developing ESL CAI at CUNY will be of use to those who want to compare CUNY programs with the proposed or existent programs at their own universities.

ESL CAI at CUNY varies widely from campus to campus. Some campuses are in the early planning stages, others are actively investigating the possibilities for CAI in ESL, and one or two are actually integrating CAI in their ESL curriculum. Many colleges are cautious about embarking on ambitious CAI programs because funding is uncertain and the field is still in a state of flux.

At Hostos Community College, Cynthia Moore (English) and Peter Martin (Student Services) report that they use a computerized interactive career planning program for students in remedial and ESL English classes. While students use the program to help define their goals, in the process they learn about the real world, become familiar with computers, and improve their English skills, especially reading and writing.

At LaGuardia Community College, Gloria Gallingane, Director of the English Language Center, has access to the microlab of 24 TRS-80 stations and is using it in a variety of ways for teachers and students. She plans to teach instructors how to use the word processor and train students to use a word processing program for composing and editing. She will use the program WordStar that has been modified by her colleague Marian Arkin (English) for upper-level students and repeaters in the ESL program. Gallingane will hire a consultant to develop individualized courseware for her program.

Clara Velazquez from Hostos (ESL) has been developing an ESL CAI computer training course for high school students who will be entering CUNY after graduation. Working out of CUNY's Office of Special Programs, Velazquez and Debra Friedman of the New York City Board of Education are in the first year of a four-year project involving several phases. The project starts with 10th grade ESL high school students who have expressed an interest in computers. The students are enrolled in ESL classes which focus on technical computer language and terminology, and their classroom instruction involves learning how to manipulate a

computer using the programming language LOGO. As their English proficiency increases in their next years of high school, so does the difficulty of their computer classes. Vocabulary used in ESL classes will be drawn from the corequisite computer classes. At later stages there will be bilingual tutoring, a guidance component, and internships involving computer use. In 12th grade, the students will be mainstreamed into regular computer courses, and in their last semester they can take a bridge course for college credit, whereupon they will be placed in a CUNY technical college (probably New York City Technical College). This program provides its students with a marketable skill, a support network, and a great deal of motivating ESL training, validating the holistic approach to CAI favored by most ESL practitioners.

ESL CAI at Bronx Community College is funded by a Title III Grant, Project Better Chance; the ESL component will run for two years under the Reading and Writing Skills Reinforcement portion. Ellen Leahy runs the microlab in which 15 or 20 Apple II Plus computers are available together with printers for hard copy. With the help of her programming assistant, Leahy programs and modifies existing courseware in response to requests from participating faculty. Irene Dutra teaches ESL with the microcomputers, and has written some programs of her own. Nineteen ESL students used the computers on a regular basis in the Spring of 1983.

Recognizing the limitations of the software in the microlab, Dutra makes the most of it by increasing student input. She has,

for instance, had some of her advanced students prepare the vocabulary items for a computer lesson for lower level students. She uses TIC-TAC-SHOW, in a game-show format in order to solicit student participation in the preparation of the items used as questions and answers in the game. She also utilizes Storyboard, a game in which the player must guess the text of a passage by asking for the placement of all instances of a letter, and repeating the process until the entire passage can be deciphered. At times, she or her students prepare the text for deciphering. Dutra also uses the Bank Street Writer word processing program to help students gain control over their writing. Her students use the lab on a regular basis.

Dutra 's approaches are the clearest in the ESL courseware she developed on "-ed" ending spelling rules. This program has a minimum of explanatory text (it is not intended to be self-standing), and pictures four columns that are filled with representative verbs "floating" into their correct slots (in earlier parts, the "-ed" ending "floats" onto the verb). At one point in the lesson using multiple choice, the student must assign a verb to the correct column; the verb soars into the correct column or is blocked from entering an incorrect one. Morse code-like beeps are used to emulate the stress pattern of the verbs (two quick beeps for the unstressed syllable, one beep for the stressed) once the correct form has landed safely. Explanations of wrong responses are not given. Eventually, the student is shown the right answer. It is possible to go forward or backward within the program.

At times, Dutra has her students working together on the computer. Two students will play TIC-TAC-SHOW, solve problems, or--in the case of some advanced students--work together authoring part of a program for other students. These activities encourage conversation among the students and help them become computer literate which helps increase their English proficiency. In turn, this experience helps the students meet the real-world needs of language use and computer literacy.

Under a 5-year program in the Department of Education at York College, Che-Tsao Huang is directing a project utilizing training workshops to educate faculty who will eventually develop their own materials. They have access to the University of Delaware's PLATO system. Peggy Eberbach (French) and Anita Wenden (Foreign Languages) are evaluating the software currently available for use in the ESL curriculum. For the time being, they are not planning to develop their own, but are utilizing commercial software. As part of her own research, Eberbach has evaluated ESL software at all levels, especially the postsecondary level, and she is an excellent resource for ESL faculty.

Michael Southwell, a pioneer in the development of basic writing software for use in York College's Comp-Lab, is now adapting his materials for ESL students, which will soon be available from a commercial publisher.

Nancy Duke Lay of City College (ESL) reports that her department is restructuring its support services for ESL students, and that "CAI will be there in addition to traditional

services." She is opposed to "quick-fix" programs, and has plans for developing ESL reading software.

At Baruch, I recently obtained funding for a microlab for ESL and have plans to develop courseware based on the research I have engaged in since early 1983, to carry out a large-scale computer-based investigation into written grammatical errors of ESL students. The aim of the research is to discover: (1) what are the most common ESL grammatical errors; (2) what are the most common ESL errors for each language group at Baruch; and (3) what are the most stigmatizing vocabulary and idiom error types (again, if possible, to discover patterns for each language group). After this we will prepare an Instructor's Guide to ESL Writing Errors in which the errors are categorized by students' first language, pinpointing trouble spots to teachers. Another project will be to prepare a lexicon of vocabulary/idiom terms with high frequency of error.

The ultimate aim is to produce a full fledged ESL CAI curriculum based on the findings of this research, with a microlab and appropriate software. At the moment, we are examining writing samples of students who fail the CUNY Writing Assessment Test (WAT). Sentences (and context, where necessary) were entered in a data-base called FOCUS on the CUNY Computer Center's mainframe operating within an IBM 370 V(irtual) M(achine) environment. The sentences are coded by first language and by error type; we hope to have a sample of 5,000 sentences. We will be able to count, call up, and examine any sentence or group of sentences by any of the parameters entered with it or in

it with FOCUS, to determine the patterns of errors with or without reference to first language. When we discover such patterns (some have begun to emerge), we hope to investigate the corresponding structure or lack of it in the student's first language so as to determine how such a mistake--either in grammar or vocabulary/idiom--could have been made. With so large a sample, and with the speed of the computer, we expect to make discoveries that are reliable, significant, and useful for remediating the errors. This kind of project can be easily adapted to a microcomputer. For further information on this project, see Dalgish (1983a and b).

The final stages of this project will incorporate interactive video with the CAI materials. In this way, we hope to address those semantic/grammatical concepts mentioned as being most troublesome to our students.

An interesting ongoing project at IBM is related to the research just described at Baruch, and although it is not a CUNY ESL project, I feel that it merits discussion here. George Heidorn and Lance Miller are constructing a parsing program called EPISTLE that can, in effect, analyze a sentence for grammatical errors. The error types that EPISTLE can recognize are of the more mechanical nature, such as subject-verb agreement, some verb forms, parallel structures, fragments, run-ons, and include some more complex error types. Although the avowed aim of the EPISTLE program is to aid the businessman in writing standard English letters, this kind of research has exciting potential for education, notably developmental English.

Whether EPISTLE will be able to decipher, much less analyze, the error types common in ESL writing, is another question. I hope that the IBM researchers will take these error types into consideration.

V. CONCLUSION

Anything one says about computer assisted instruction is bound to be obsolete almost as soon as it is written, yet I would like to leave the reader with several thoughts at this time. My main concerns are the current state of the art, the need for networking and communication, the importance of careful thought and articulation of overall philosophies toward CAI and ESL, and possible future directions.

One message is clear--there is very little high quality ESL software currently available. (A participant at the end of a recent conference asked if there was anything in ESL CAI that anyone liked; the question was met with total silence.) Improvement will occur only after teachers begin programming on their own. This activity is very time-consuming, and since a newly funded computer lab must be functioning immediately, the tendency is to purchase what is commercially available. This proliferates poor quality software. Extensive networking may help avoid this trap, but the bottom line is that most commercial software will not fit the needs of every student or every teacher.

In-house production of software, however, raises other problems. It is doubtful whether it can advance an academic career, and it may often prove wasteful in its reinvention of the wheel. Instructors-turned-programmers may find themselves devoting months on a project which a colleague at another institution has completed. To avoid this will require extensive networking formally and informally and a willingness to share research. Lest this last point sound too naive, some presenters at CUNY's 1984 "Microcomputers and Basic Skills in College" conference did just that--they demonstrated and offered in-house produced software for public use. It is unlikely that programmers in ESL will make a fortune from their own courseware anyway, so it may as well be in the public domain.

We also find very little discussion of the philosophical underpinnings of ESL CAI. What are we trying to do with CAI and in what framework are we trying to do it? For many, these questions are raised after the work has begun, and even for those who ask, the answers seem to shift as they go further along into the field. There is a need for a paradigm that is specific to CAI ESL, a paradigm that is also applicable to current trends in ESL.

The fields of artificial intelligence and automated parsing have much to teach and much to learn from the ESL field. It will not be long before syntactic diagnosis of a student's writing will be accomplished by parsers, and speech recognition devices will be available to help ESL students in production practice. Moreover, the potential of interactive video (tape or disk) has

already been recognized, as more and more applications of this resource are developed and modified for use in ESL.

The future is here for microcomputers in the ESL classroom, and yet there is still much to do. We must tell each other of our proposals, our plans, our programs, our dreams, and our philosophies, lest we fail to fulfill the potential that the new educational technology offers us.

VI. GLOSSARY

- Authoring** Writing, or participating in preparing, software or lessons for CAI. Computer language like BASIC, LOGO, PASCAL, and LISP are the major languages used in authoring. An authoring program or package is software that provides the structure of the lesson and the author simply plugs in his own material, as in PILOT.
- BASIC** An acronym for the computer language Beginner's All-Purpose Symbolic Instruction Code. Developed by Kemeny and Kurtz at Dartmouth in the 1960s. Generally considered easy to learn and commonly taught as a first language.
- Bit** Short for binary digit, the smallest unit of information in a computer's memory. Eight bits = a BYTE.
- Booting Up** Getting started on the computer. For micros it means either inserting the diskette with the computer off, then turning it on, or using one of the BASIC commands to have the computer read the diskette inserted in the disk drive. See LOAD. Said to have taken its name from the idea of lifting oneself up by one's bootstraps.
- Branching** The ability of software or courseware to take alternative paths through a lesson depending on (student) input. In practice, it means that the user is not lockstepped along one and only one path through the lesson. It is also loosely used to mean that alternative answers (output) are given depending on the responses the user makes when asked a question.
- Bug** A problem, mistake, or error in the computer or the program. Most are caused by users, but some software has its own share of built-in problems. The hardware is the least likely culprit responsible for a bug, but it does happen. Said to have received its name from the time when an actual bug was found in the innards of a computer causing it to malfunction (probably apocryphal). See DEBUG.

BYTE	A group of eight BITS. The usual home computer or micro contains BYTES by the thousands, as in KILOBYTE.
Cassette Player	A tape recorder that substitutes for a disk drive, with the software information on a cassette tape instead of a disk. The cassette system interfaces with the computer, but is subject to error so much that disk drives are usually preferred. A portable cassette player will not work with most computers; you have to buy the manufacturer's "special" machine.
CMI	Abbreviation for <u>computer managed instruction</u> , instruction in which the computer provides overall management, as in testing, evaluation, record-keeping, planning, etc. All CAI has some elements of CMI in it.
Courseware	CAI lessons or software, a term often used to distinguish between instructional software and non-instructional materials.
CP/M	Abbreviation for <u>Control Program for Micro-processors</u> , a program that tells computers how to get information from disks (see DOS). Most business software is written in CP/M, but most educational material is not.
CPU	Abbreviation for <u>central processing unit</u> , the processor of a computer. On micros, this is the integrated circuit, the brains of the machine.
CRT	Abbreviation for <u>cathode ray tube</u> , the T.V. monitor that displays the information from the computer.
Cursor	The blinking rectangle or square on a CRT that tells you where you are on the screen. When the cursor stops blinking, it usually means trouble.
Debug	To remove the BUGs from a system.
Default value	The option or direction a program will take if the user does not indicate a special, unusual choice.
Disk (Diskette)	A flexible piece of plastic covered with iron oxide (sometimes known as rust) that

is used to store information. They come in three sizes: 5 1/4 in., 8 in., and a new 3 1/2 in. The disk is alternately read to or written from the computer's memory by a DISK DRIVE, which spins the disk at several hundred rotations per minute to access the information. You have to know if your computer's disk drive is double-sided or single-sided (has one or two read/write heads) so that the disks you buy are the right kind. Disks are double density and single density, too. See also HARD DISK.

Documentation

Computerese for the written material that comes with your computer or the software you buy. If you intend to produce your own software, you will need to provide readable documentation. Most manuals are very poorly written (either too cutesy or too technical).

DOS

Abbreviation for Disk Operating System, the system that tells the computer that load means load, save means save, write means write, etc. The DOS for many manufacturers is not the same, and since software is written in only one DOS at a time, you have to be sure your computer can read it.

Down

The condition in which the computer or computer system is not working because of a failure, usually when one is right in the middle of an important job.

File

The name given by the user for a collection of data. Most programs need file names so the computer can fetch what you want. As a verb, it refers to keeping the changes you may have made to a file in memory; it is nearly synonymous with SAVE.

Format

The structure or layout of instructional material. As a verb, it refers to the process of preparing a disk so that it may receive data (A totally blank disk can't do anything.)

Frame

The amount of material on the screen at any one time.

GIGO	Acronym for "garbage in, garbage out", a phrase referring to the fact that the quality of what the computer produces as output is only as good as the quality of its input.
Graphics	Pictures, illustrations or charts on a screen, that may or may not move around.
Hard copy	A printout of information, not always the same as what appeared on your screen but usually near enough.
Hard disk	A metallic disk, usually of aluminum, used instead of a FLOPPY. It can hold MEGABYTES of information, is spun at speeds of 30 revolutions per second, and is sometimes more expensive than a computer itself.
Hardware	The computer, the disk drive, the printer, or various peripherals excluding the software.
Help Menu	A set of instructions on the screen (or available by pushing a HELP button on some computers) that will assist the user by explaining various options if he is unsure of what command to use. See MENU.
High-level	In a computer language, closer to English, often utilizing real words like "run", "print", and some hybrids like "goto."
Input	Student or user response to the computer.
Interactive	Allowing for the student and the system to alternate in addressing one another, so that responses depend on each other, and usually more than simply "yes/no" or "T/F".
Interactive Video	Computer controlled videotape or videodisc used in courseware.
Interface	Hardware or software that links systems, devices, or the like. As a verb, to make sure connections or links.
K or Kilobyte	1024 bytes of memory, a typical unit of storage. A double-spaced typewritten page takes somewhere between 2 or 3 Kilobytes.

Load	To take from storage or disk and bring to the computer's memory to be displayed on the screen for editing.
Lock	To protect a file from being deleted, undergoing a name change, or being changed in most ways. Sometimes software is said to be locked in that it cannot be altered by the user. Compare with PROTECT.
LOGO	A higher-level interactive programming language developed at MIT and designed for use by students.
Low-level	In a computer language, close to the level of the electronic components of the computer. See MACHINE LANGUAGE; HIGH-LEVEL.
Machine language	The language of the machine, the coding system for numbers, letters, etc. The computer does not have to translate anything written in this language in order to operate it.
Mainframe	The most powerful type of computers, capable of transmitting 10 million instructions per second.
Menu	A set of instructions for running a program, displayed on the screen (or available by pushing a button), allowing the user to merely select among options. It should be easy to get to a menu from within a program. A program that is menu-driven is one that, at every step, has a menu forcing the user to choose among set alternatives. For some, this is too confining; for others, it is a relief. See HELP MENU
Micro	Short for microcomputer, or personal computer. Most can deliver 250,000 instructions per second. Compare MAINFRAME; MINI. In education, especially in ESL, the most popular brand is the Apple II Series.
Mini	Short for minicomputer, a computer more powerful than a micro, but usually less powerful than a mainframe.

Mini-authoring	Producing lessons in a restricted format, often no more than dittos on computer, with multiple choice or true-false type exercises predominating.
Modem	A device that connects a terminal to a computer via a telephone.
Monitor	A display unit similar to a television screen but with higher resolution. Often in green phosphor; color varieties are very expensive.
PILOT	An acronym for Programmed Inquiry, Learning or Teaching. A high level authoring language used especially for courseware.
PLATO	An acronym for Programmed Logic for Automated Teaching Operations. A nationwide software delivery system involving mainframes. Developed in 1959 at the University of Illinois. It offers access to a large number of college level programs. Some PLATO software is now available for use on micros.
Prompt	A word or technique built into the lesson to tell the user what to do.
Protect	To electronically modify software in order to prevent any change made to the disk. As a result, the disc or program cannot be copied.
RAM	An acronym for <u>Random Access Memory</u> , memory that can be altered, added to, erased or examined. Sometimes also called user programmable memory. RAM vanishes when the computer is turned off, hence the necessity for BACKUP. Compare ROM.
ROM	An acronym for <u>Read Only Memory</u> , memory that the computer can read but that the user cannot alter. ROM is placed in the computer permanently by the manufacturer and cannot be erased even when the current is turned off.
Run	An instruction to the computer to execute a program or set of instructions.
Save	A command to keep what you have typed and store it permanently on a disk or in a

file. If you forget to save what you have typed, you could lose all your work. In some systems, this command is very similar to FILE.

Scroll

The manner in which material is moved on a screen, resembling the unravelling of an imaginary piece of paper, with characters at the top vanishing as new material is added at the bottom.

Software

Instructions, or pre-packaged programs on disks to be run on the computer. Educational lesson material or software is sometimes referred to as COURSEWARE.

Time-sharing

A system in which a number of users simultaneously access a computer, while the computer divides up its work and time among the users. PLATO is an example of a time-sharing system.

Unlock

To remove the protection of a file that was locked, thereby allowing it to be altered, erased, etc.

User-friendly

Aimed at being simple to use or control; providing clear directions.

VII. REFERENCES

- Alesandrini, Kathryn L. 1982. "CAI Should Differ from Tradition." Electronic Education 2(2):15,22-3.
- Barger, Robert Newton. 1982. "The Computer as a Humanizing Influence in Education." T.H.E. Journal 9(4):95-6, 105
- Baum, Joan. 1983. Computers in The English Class: With Particulars Attention To The City University of New York. New York: Instructional Resource Center, Office of Academic Affairs, The City University of New York.
- Bork, Alfred. 1981. "Educational Technology and the Future." Journal of Educational Technology Systems 10(1):3-20.
- Boyd, Gary., et al. 1982. "Remedial and Second Language English Teaching Using Computer-Assisted Learning." Computers and Education 6(1):105-112.
- Braun, Ludwig. 1980. "Computers in Learning Environments: An Imperative for the 1980's." Byte 5(7):6-10,108-114.
- Burke, Robert L. 1982. CAI Sourcebook. Englewood Cliffs, NJ: Prentice-Hall Inc.
- Burns, Hugh L. and George H. Culp. 1980. "Stimulating Invention in English Composition Through Computer-Assisted Instruction." Educational Technology August, 1980:5-10.
- Caldwell, Robert M. 1980. "Guidelines for Developing Basic Skills Instructional Materials for Use with Microcomputer Technology." Educational Technology October, 1980:7-12.
- Cherry, Lorina. 1981. "Computer Aids for Writers." Bell Labs.
- Coburn, Peter et al. 1982. Practical Guide to Computers in Education. Redding, MA:Addison-Wesley.
- Dalgish, Gerard M. 1983a. "ESL Error Analysis and Individualized CAI." Proceedings of the Microcomputers and Basic Skills Conference, CUNY and LIU. Instructional Resource Center, CUNY.
- _____. 1983b. "Dictionary Selection and the Lexicon." Dictionaries; Proceedings of the Dictionary Society of North America Biennial Meeting.
- Daynes, Rod. 1982. "The Videodisc Interfacing Primer." Byte 7(6): 48-59.

- DeBloois, Michael L. 1982. Videodisc/Microcomputer Courseware Design. Englewood Cliffs, NJ. Educational Technology Publications.
- Dixon, Rebecca. 1981. "PLATO Reaches International Students with English Lessons." Studies in Language Learning 3(1):98-112.
- Dodge, James W. 1980. "Educational Technology." In Geno:99-119.
- English, Randall. 1983. "Problems in Paradise: We Will Be Disappointed with the Computer." Electronic Education. 2(5): 24, 38.
- Gallagher, Brian. 1984. Microcomputer Word Processing Programs: An Evaluation and Critique. Instructional Resources Center, Office of Academic Affairs, The City University of New York (forthcoming)
- Geno, Thomas H. (ed.) 1980. Our Profession: Present Status and Future Directions. Middlebury, Vermont.
- Hammond, Allen L. 1972. "Computer-Assisted Instruction; Many Efforts, Mixed Results." Science 176:1025-6.
- Harrison, John S. 1983. "Foreign Language Computer Software: What? Where? How Good?" Newsletter of the Northeast Conference on the Teaching of Foreign Languages 13, 2/83: 26-30.
- Hart, Robert S. (ed.) 1981. Studies in Language Learning: Special Issue on the PLATO System and Language Study 3.1.
- Heidorn, George E. et al. 1982. "The EPISTLE Text-Critiquing System." IBM Systems Journal 21(3):305-321.
- Higgins, John and Tim Johns. 1984. Computers in Language Learning. Redding, MA: Addison-Wesley.
- Howe, J. and B. DuBoulay. 1979. "Microprocessor-assisted Learning: Turning the Clock Back?" Programmed Learning and Educational Technology 16(3):240-6.
- Jamieson, Joan, and Carol Chapelle. 1980. "ESL Computer-Assisted Instruction on PLATO." ms., University of Illinois, Division of ESL.
- _____. 1983. "Using Informative Feedback Messages in CALL Courseware." TESOL Newsletter 8/83:26-27.
- Jorstad, Helen. 1980. "New Approaches to the Assessment of Language Learning." In Geno:121-140.
- Kaplan, Robert B. 1970. "Cultural Thought Patterns in Inter-Cultural Education." Language Learning 16:1-2.

- Lavine, Roberta Z. 1981. "Computer-Assisted Instruction in the ESL Curriculum." ERIC, #ED214391 FL012844.
- Marty, Fernand. 1981. "Reflections on the Use of Computers in Second-Language Acquisition." Studies in Language Learning 3(1):25-53.
- McWilliams, Peter A. 1982. The Personal Computer Book. Los Angeles:Prelude Press.
- Orwig, Gary. 1983. Creating Computer Programs for Learning. Reston, VA.:Reston Publishing.
- Otto, Frank. 1980. "Computer-Assisted Instruction (CAI) in Language Teaching and Learning." In Annual Review of Applied Linguistics, R. Kaplan, R. Jones and G. Tucker, eds, Rowley, MA.:58-69.
- Papert, Seymour. 1980. Mindstorm: Children, Computers and Powerful Ideas. New York: Basic Books, Inc.
- Saracho, Olivia. 1982. "Planning Computer-assisted Instruction for Spanish-speaking Migrant Students." Journal of Educational Technology Systems 10(3):257-260.
- Smith, Stanley C. and Bruce Arne Sherwood. 1976. "Educational Uses of the PLATO Computer System." Science 192:344-352.
- Southwell, Michael G. 1982a. "Computers and Developmental Writing at York College/CUNY." The Computing Teacher Nov., 1982:34-35.
- _____. 1982b. "Using Computer-Assisted Instruction for Developmental Writing." Association of Educational Data Systems Journal Winter, 1982:80-91.
- Stevens, Vance. 1980. "The CAL Facility at UPM." Teachers of English Arabic Monthly 37:12-15.
- _____. 1983. "English Lessons on PLATO." TESOL Quarterly 17.2:293-300.
- The, Lee. 1982. "Educational Software for the Home." Personal Computing 6(6):48-52,102-114.
- Wyatt, David. 1983. "Computer-Assisted Language Learning Comes of Age." TESOL Quarterly 6/83:9.
- _____. 1984. Language Learning and Computers: A Practical Sourcebook. New York, NY:Regents.

VIII. SOURCES AND RESOURCES

- American Language Academy, Suite 200, 11426 Rockville Pike,
Rockville, MD 20852
- Apple Educator's Newsletter, 9525 Lucerne, Ventura, CA 93004
- Apple Education News, 10260 Bandley Drive, Cupertino, CA 95014
- AppleWriter, Apple Computer Corporation, Inc., 20525 Mariani
Avenue, Cupertino, CA 95014
- Arkin, Marian, English, LaGuardia Community College
- Bank Street Writer (available from), Scholastic Inc., 904 Sylvan
Avenue, Englewood Cliffs, NJ 07632
- Baum, Joan, English, York College
- BIPACS, 33 W. Walnut Street, Long Beach, NY 11561 (markets
software)
- Brandeau, John, Audio-Visual, Bernard M. Baruch College
- Byte, 70 Main Street, Peterborough, NH 03458
- CALICO (Computer Assisted Language Learning Instruction
Consortium), 229 KMB, Brigham Young University, Provo, UT
84602
- CBS Professional Publishing/Holt, Rinehart, and Winston, 383
Madison Avenue, New York, NY 10017
- Classroom Computer News, PO Box 266, Cambridge, MA 02138
- COMPRESS, P.O. Box 102, Wentworth, NH 03282 (markets software)
- Compute!, 515 Abbot Drive, Broomall, PA 19008
- Computer Curriculum Corporation, 1070 Arastradero Road, Palo Alto
CA 94304 (mini-computer in ESL)
- Computers in the Media Center, CMC News, 515 Oak St. N., Cannon
Falls, MN 55009
- Computing Teacher, Dept. of Computer Science, U of Oregon,
Eugene, OR 97203
- Control Data Corporation, 2000 West Loop South, Houston, TX 77027
(markets the PLATO system)
- Creative Computing, 39 East Hanover Ave., Morris Plains, NJ 07950
- DORMAC, PO Box 752, Beaverton, OR 97075 (markets software).

Dorsett Educational Systems, Inc., PO Box 1226, Norman, OK 73070
(markets software).

Dutra, Irene, Special Education, Bronx Community College

Eberbach, Peggy, Foreign Languages, York College

Educational Computer, PO Box 535, Cupertino, CA 95015

Educational Technology, ET Publications, Englewood Cliffs, NJ
07632

Electronic Education, 1311 Executive Center Drive, Tallahassee,
Florida 32301 (or: PO Box 20221, ZIP 32304)

Electronic Learning, 902 Sylvan Ave., Englewood Cliffs, NJ 07632

Epie Report, Epie Institute, Box 620, Stony Brook, NY 11790

EPISTLE, IBM, The T.J. Watson Research Center, Box 218, Yorktown
Heights, NY 10598

FOCUS, Information Builders Inc., 1250 Broadway, New York,
NY 10001

Franklin AceWriter, Franklin Computer Corporation, 2128 Route 38,
Cherry Hill, NJ 08034

Gallingane, Gloria, English Language Center, LaGuardia Community
College

Gallagher, Brian, English Department, LaGuardia Community College

Hands On! TERC, 8 Eliot Street, Cambridge, MA 01238

Hartley Courseware, Inc., Box 431, Dimondale, MI 48821

Heidorn, George, IBM, The T.J. Watson Research Center, Box 218,
Yorktown Heights, NY 10598

Hiatt, Mary, English, Bernard M. Baruch College

Homeword, Sierra On-Line Building, Coarsegold, CA 93614

Huang, Che-Tsao, Department of Education, York College

Infoworld, 375 Cochituate Rd., Framingham, MA 01701

Interface Age, 16704 Marquardt Ave., Ceritos, CA 90701

Journal of Computer-Based Instruction, ADCIS International Head-
quarters, Computer Center, Western Washington University,
Bellingham, WA 98225

Journal of Courseware Review, Box 28426, San Jose, CA 95159

Kilobaud Microcomputing, PO Box 997, Farmingdale, NY 11737

Lay, Nancy Duke, English as a Second Language, The City College

Leahy, Ellen, Learning Specialist, Bronx Community College

LOGO Computer Systems, Inc., 368 Congress St., Boston MA 02210
(markets software and hardware using LOGO)

Martin, Peter, Student Services, Hostos Community College

McDermott, Robert, Philosophy, Bernard M. Baruch College,

MEAN BRIEF, Microcomputer Education Applications Network, 256
North Washington Street, Falls Church, VA 22046

MECC (Minn. Educational Computer Consortium), 2520 Broadway Dr.,
St Paul, MN 55113

Microcomputer Index, Microcomputer Information Services, 2464 El
Camino Real, Suite 247, Santa Clara, CA 95051

Microcomputers in Education, Queue, Inc., 5 Chapel Hill Drive,
Fairfield, CT 06432

Microform Review, 520 Riverside Ave., Westport CT 06880

MICRO-SCOPE, JEM Research, Discovery Park, University of Victoria
Box 1700, Victoria, B.C. V8W2Y2

Microsift, Northwest Regional Educational Lab., 300 S.W. Sixth
Ave., Portland, OR 97204

Miller, Lance, IBM, The T.J. Watson Research Center. Box 218,
Yorktown Heights, NY 10598

Moore, Cynthia, English, Hostos Community College

Nibble, 11021 Magnolia Blvd., North Hollywood, CA 91601

Northeast Conference on the Teaching of Foreign Languages,
Newsletter, Box 623, Middlebury, VT 05753

Peelings II, PO Box 188, Dept. N, Las Cruces, NM 88004

Personal Computing, 4 Disk Drive, Box 10408, Riverton, NJ 08077

Pipeline, Conduit, PO Box 388, Iowa City, IA 52244

Popular Computing, Byte Publications (q.v.)

PLATO, University of Delaware, Newark, DE 19711

- PLATO, University of Illinois, contact Robert Hart, Foreign Languages Building, 408 South Matthews, University of Illinois, Urbana, IL 61801
- QUEUE, 5 Chapel Hill Drive, Fairfield, CT 06432
- Radio Shack Educational Division, Dept. 82A 356, 1300 One Tandy Center, Ft. Worth, TX 76102
- Regents/American Language Academy (ALA) Company, 2 Park Avenue, New York, NY 10016
- Santillana Publishing Co. Inc, 257 Union Street, Northvale, NJ 07647
- Seliger, Herbert, Linguistics, Queens College
- Small, University Station, Brigham Young University, Box 7134, Provo, UT 84602
- Softalk, Softalk Circ., 11021 Mangnolia Blvd., North Hollywood, CA 91601
- Southwell, Michael, English, York College
- Stevens, Martin, Dean, Liberal Arts and Sciences, Bernard M. Baruch College
- Storyboard (available from) CBS Professional Publishing/Holt, Reinhart, and Winston, 383 Madison Avenue, New York, NY 10017. Also available from WIDA Software, 2 Nicholas Gardens, London W5 FHY England
- Teacher's Friend, The Soft Spot, 800 East Arapaho, Suite 110, Richardson, TX 75081
- T.H.E. Journal (Technological Horizons in Education), PO Box 992, Acton, MA 01720
- TICCIT, a mini-computer facility with ESL. (Dr. Frank Otto) See CALICO.
- Tic-Tac-Show, Computer Advanced Ideas, 1442A Walnut Street, Suite 341, Berkeley, CA 94709
- Tilyer, David, Regents/ALA Company, 2 Park Avenue, New York, NY 10016
- Velazquez, Clara, ESL, Hostos Community College
- Wendon, Anita, Foreign Languages, York College
- WIDA Software, 2 Nicholas Gardens, London, W5,FHY, England
- WordStar, Micropro International, 33 San Pablo Avenue, San Rafael, CA 94901

c Copyright 1984 by the Office of Academic Affairs
The City University of New York